

Attorney Docket No.: 2489.P002DC

PATENT

UNITED STATES PATENT APPLICATION

For

SYNCHRONOUS CONTENT ADDRESSABLE MEMORY

INVENTORS:

VARADARAJAN SRINIVASAN  
BINDIGANAVALE S. NATARAJ  
SANDEEP KHANNA

Prepared By:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP  
12400 WILSHIRE BOULEVARD  
SEVENTH FLOOR  
LOS ANGELES, CA 90025-1026

(408) 720-8300

"Express Mail" mailing label number: EV409362440US

Date of Deposit: 12/22/03

I hereby certify that I am causing this paper or fee to be deposited with the United States Postal Service "Express Mail Post Office to Addressee" under 37 C.F.R. § 1.10 on the date indicated above and that this paper or fee has been addressed to the Commissioner for Patents, PO Box 1450, Alexandria, Virginia 22313-1450.

JUANITA BRISCOE

(Typed or printed name of person mailing paper or fee)

(Signature of person mailing paper or fee)

12/22/03

(Date signed)

## SYNCHRONOUS CONTENT ADDRESSABLE MEMORY

### CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation application of U.S. Patent Application No. 09/778,170 entitled "SYNCHRONOUS CONTENT ADDRESSABLE MEMORY," filed on February 6, 2001, which is a divisional application of U.S. Patent 6,199,140, filed on October 30, 1997.

### FIELD OF THE INVENTION

[0001] The present invention relates generally to content addressable memory (CAM) devices.

### BACKGROUND

[0002] A content addressable memory (CAM) device is a storage device that can be instructed to compare a specific pattern of comparand data with data stored in its associative CAM array. The entire CAM array, or segments thereof, are searched in parallel for a match with the comparand data. If a match exists, the CAM device indicates the match by asserting a match flag. Multiple matches may also be indicated by asserting a multiple match flag. The CAM device typically includes a priority encoder to translate the matched location into a match address or CAM index and outputs this address to a status register.

[0003] Each CAM cell conventionally includes a comparator and a random access

memory (RAM) element. The CAM array may be partitioned into separate segments in which one segment stores CAM or compare data to be compared with the comparand data, and another segment stores associated RAM data corresponding to each of the CAM or compare locations. Once a match between the CAM data and the comparand data is determined, the associated RAM data for the matched location may be output to a status register. The RAM data and/or the CAM data may then be read from the status register.

**[0004]** Conventional CAM devices require more than one clock cycle to perform a write and compare instruction. For example, a typical write and compare instruction requires at least three clock cycles: a first clock cycle to present a compare instruction and/or comparand data to the CAM device, perform the search, and generate a match flag and multiple match flag signal; a second clock cycle to instruct the CAM device to output the matching CAM address or index; and, a third clock cycle to instruct the CAM device to output the associated data and status information (e.g., skip bit, empty bit, full flag, as well as, match and multiple match flags) for the matched location. With conventional cycle times generally running at 100 nanoseconds (ns), it requires at least 300 ns to complete this process. This generally limits the search rate of conventional CAM devices to approximately 1 to 3 million searches per second. This also generally limits the number of ports, segments, or devices that can be supported by a conventional CAM device in a switch or router environment.

**[0005]** The multi-clock cycle process has generally been required due to the architecture of conventional CAM devices. Most CAM devices include a general purpose bi-directional bus that keeps the pin count of the CAM devices to a minimum (e.g., 44

pins). The bi-directional bus is used to load comparand data and instructions into the CAM device. The bi-directional bus is also used to output the matched address, associated data, and status bits from a status register in the CAM device. Because this bus is shared with so many input and output functions, it requires many clock cycles to multiplex data on the bus.

[0006] CAM devices that have separated the general purpose bi-directional bus into a data input bus and a data output bus, still require at least three clock cycles to perform the write and compare operation described above, namely: one clock cycle to load the write and compare instruction and/ or load the comparand data and perform the comparison with CAM array; one clock cycle to access the associated data; and, one clock cycle to instruct the CAM device to output the match address, associated data, and/ or status information.

[0007] As applications for CAM devices increase in speed, there has been a desire for faster CAM devices that have shorter search times, or preferably, can execute a write and compare instruction in a fewer number of clock cycles. For example, it is desirable to have a CAM device that can be used as an address filter or address translator in an ethernet switch or router that operates at data rates of 100 Megabits per second (Mb/s) to 1 Gigabits per second (Gb/s). It is also desirable to have a CAM device that can be used to implement fast routing tables in Internet Protocol (IP) switches. As the number of ports, segments, or devices that are supported by the switches or routers increases, the time required for the supporting CAM device to perform a write and compare operation (e.g., address filter or translation operation) decreases. For example, to support a 1 Gb/s

ethernet switch, a CAM device supporting approximately three ports should advantageously be able to perform a single write and compare instruction in approximately 100 ns or faster. A CAM device supporting approximately six ports should advantageously be able to perform a single write and compare instruction in approximately 50 ns or faster.

## SUMMARY OF THE INVENTION

[0008] A content addressable memory (CAM) device is disclosed. The CAM device is a synchronous device that may perform all of the following operations in one clock cycle: (1) receive comparand data from a comparand bus; (2) receive an instruction from an instruction bus instructing the CAM device to compare the comparand data with a first group of CAM cells in a CAM array; (3) perform the comparison of the comparand data with the first group of CAM cells; (4) generate a match address for a location in the CAM array that stores data matching the comparand data; (5) access data stored in a second group of CAM cells in the CAM array, wherein the second group of CAM cells may store data associated with the matched location; and (6) output to an output bus the match address, the data stored in the second group of CAM cells, and/or status information corresponding to the matched address or the second group of CAM cells. The status information may include a match flag, multiple match flag, full flag, skip bit, empty bit, or a device identification for the CAM device.

[0009] Other objects, features, and advantages of the present invention will be apparent from the accompanying drawings and from the detailed description which follows below.

## BRIEF DESCRIPTION OF THE DRAWINGS

[0010] The features and advantages of the present invention are illustrated by way of example and are by no means intended to limit the scope of the present invention to the particular embodiments shown, and in which:

[0011] Figure 1 is a block diagram of one embodiment of a CAM device according to the present invention;

[0012] Figure 2 is block diagram of one embodiment of the CAM array of Figure 1;

[0013] Figure 3 is a block diagram of one embodiment of a CAM cell;

[0014] Figure 4 is one embodiment of the CAM cell of Figure 3;

[0015] Figure 5 is one embodiment of a method of performing a write and compare operation in a single clock cycle;

[0016] Figure 6 is a timing diagram for one embodiment of the signals generated by the timing generator of Figure 1;

[0017] Figure 7 is one embodiment of the timing generator of Figure 1;

[0018] Figure 8 is another embodiment of the timing generator of Figure 1;

[0019] Figure 9 is a timing diagram illustrating a pipelined mode of operation for the CAM device of Figure 1;

[0020] Figure 10 is one embodiment of the signals output onto the ADS BUS of Figure 1; and

[0021] Figure 11 is another embodiment of the signals output onto the ADS BUS of Figure 1.

## DETAILED DESCRIPTION

**[0022]** A content addressable memory (CAM) device is disclosed. In the following description, for purposes of explanation, specific nomenclature is set forth to provide a thorough understanding of the present invention. However, it will be apparent to one skilled in the art that these specific details may not be required to practice the present invention. In other instances, well known circuits and devices are shown in block diagram form to avoid obscuring the present invention unnecessarily. Additionally, the interconnection between circuit elements or blocks may be shown as buses or as single signal lines. Each of the buses may alternatively be single signal lines, and each of the single signal lines may alternatively be buses.

**[0023]** The CAM device of the present invention is a synchronous device that has an instruction bus for receiving instructions, a separate comparand bus for receiving comparand data to be compared with one or more CAM cells of a CAM array, and a separate associated data and status bits bus. The associated data and status bits bus may simultaneously or individually output: a match address or CAM index for a location of the CAM array that matches the comparand data; data stored in one or more of the CAM cells of the CAM array, wherein the data is associated with the matched address; and status information corresponding to the matched address or associated data. The status information may include a match flag, multiple match flag, full flag, skip bit, empty bit, or device identification information for the CAM device.

**[0024]** The CAM device may perform, in a single clock cycle (i.e., a flow through mode), a write and compare instruction that causes the CAM device to: (1) receive



comparand data from the comparand bus; (2) receive an instruction from the instruction bus instructing the CAM device to compare the comparand data with a first group of CAM cells in a CAM array; (3) perform the comparison of the comparand data with the first group of CAM cell(s); (4) generate a match address if a location in the CAM array stores data matching the comparand data; (5) access data stored in a second group of CAM cells corresponding to the matched location in the CAM array, wherein the accessed data is associated with the matched address; and (6) output the matched address, the data stored in the second group of CAM cells, and/ or the status information to associated data and status bits bus.

**[0025]** The single cycle may have any cycle time. For one embodiment, the single cycle time may be approximately 25 ns and the CAM device may have a search rate of approximately 35 to 45 million searches per second. Thus, the present invention may be very useful in a high-speed (e.g., 100 Mb/s or 1 Gb/s) network bridge or router environment. For example, the CAM device of the present invention may support eight or more ports, segments, or devices in a high-speed ethernet switch or router environment having, for example, a data rate of 1 Gb/s. The CAM device of the present invention may store destination addresses of data packets sent between the segments, ports, or devices in the network.

**[0026]** Figure 1 shows CAM device 100 according to one embodiment of the present invention. CAM device 100 includes three separate ports coupled to three separate buses. The first port is coupled to a comparand bus (CBUS) 138 that may be a bi-directional bus used to provide comparand data to comparand register 120. CBUS 138 may also be used

to access device configuration register 136, status register 132, device identification register 121, memory configuration register 106, CAM 102, and one or more mask registers (not shown). CBUS 138 may be any size to accommodate any number of bits. For one embodiment, CBUS 138 is a 64-bit bus. The first port may have input buffers or registers coupled to CBUS 138.

**[0027]** The second port is coupled to an instruction bus (IBUS) 140 that is used to provide instructions to instruction decoder 128. Instructions may be clocked into instruction decoder 128 from IBUS 140 by one or more clock signals output from clock buffer 124 to bus 144. IBUS 140 may be any size to accommodate any number of bits and any number of instructions. For one embodiment, IBUS 140 is 14 bits wide to accommodate 2<sup>14</sup> unique possible binary coded instructions. Other encodings may be used. The second port may have input buffers or registers coupled to IBUS 140.

**[0028]** Clock buffer 124 may buffer the external clock signal CLK 178 and provide one or more clock signals to timing generator 126 via bus 180, and may provide one or more clock signals to instruction decoder 128 via bus 144. Clock buffer 124 may also generate clock signals having varying phases and frequencies.

**[0029]** The third port is coupled to an associated data and status bit bus (ADS BUS) 142 that may output matched address data, data from CAM array 104 corresponding to the matched address, and/or status information. ADS BUS 142 may be any size to accommodate any number of bits. For one embodiment, ADS BUS 142 is a 64-bit bus. The third port may have output buffers or registers coupled to ADS BUS 142.

**[0030]** Output multiplexer 134 provides data to ADS BUS 142. Output multiplexer

134 may include output buffers, one or more multiplexers, a selector circuit, registers, or latches. Output multiplexer 134 may receive a matching CAM address or index from priority encoder 116 via bus 146, and may also receive data stored in CAM array 104 via sense amplifiers 122 and bus 152. Additionally, output multiplexer 134 may receive status information including a match flag signal (MF) from priority encoder 116 via line 148, a multiple match flag signal (MMF) from priority encoder 116 via line 150, a full flag signal (FF) from flag logic 130 via line 156, device identification information from device identification register 121 via bus 155, and/or validity bits 108 via bus 157. The status information will be described in more detail below. Output multiplexer 134 may alternatively or additionally receive the matching CAM address, data stored in CAM array 104 corresponding to the matching CAM address, and/or the status information from status register 132 via bus 158. Configuration register 136 may store one or more programmable bits that may control whether output multiplexer 134 outputs the matching CAM address, CAM array data, and status information from status register 132 (e.g., in a pipelined mode) or from the other circuit elements (e.g., in a single cycle flow through mode). Status register 132 may comprise one or more registers.

**[0031]** CAM device 100 may also include flag logic 130 that may generate a full flag (FF) signal on lines 156 in response to validity bits 108 on bus 157. Flag logic 130 may also generate a match flag signal and a multiple match flag signal on lines 156. The flag(s) may be coupled to output multiplexer 134 and/or status register 132.

**[0032]** CAM device 100 may also include device identification register 121 that may store device identification information that identifies CAM device 100 from other CAM

devices in a system. The device identification information may comprise any number of bits or signals. For one embodiment, the device identification information is 16 bits of binary encoded information. Any other encoding format may be used. The device identification information may also be provided to status register 132.

[0033] CAM device 100 also includes CAM 102. CAM 102 includes a CAM array 104 that may be organized in any number of rows and columns of CAM cells. CAM 102 may also include validity bits 108 that store information about corresponding locations in CAM array 104. For example, the validity bits for a given row or location in CAM array 104 may include a skip bit and an empty bit. The skip bit may indicate that a particular location in CAM array 104 should be skipped when performing a compare operation with comparand data stored in comparand register 120. The empty bit may indicate that a corresponding location in CAM array 104 is empty. The validity bits also group the CAM cells into four states as indicated in Table 1. A comparison operation may compare comparand data against any of the locations in CAM array 104 that correspond to a given state.

Table 1

SKIP	EMPTY	STATE
0	0	VALID
0	1	EMPTY
1	0	SKIP
1	1	RAM

**[0034]** Validity bits 108 may be provided (e.g., through sense amplifiers) to output multiplexer 134, status register 132, and/or flag logic 130 via bus 157. Alternatively, validity bits 108 may be generated or decoded in response to a decoded instruction output by instruction decoder 128. Figure 2 shows one embodiment of CAM array 102 having a plurality of CAM cells 202 organized in any number of rows and columns. For one embodiment, CAM array 102 may include approximately 4k (i.e., 4086) rows and approximately 64 columns of CAM cells 202. For another embodiment, CAM array 102 may include approximately 2k (i.e., 2048) rows and approximately 128 columns of CAM cells 202. The validity bits may also be included in additional rows and columns of CAM cells 202.

**[0035]** Each row of CAM cells 202 is coupled to a match line 204 and a word line 208. Each word line 208 is driven by address decoder 112 to select one or more of CAM cells 202 for writing or reading. Each match line 204 is coupled to match latch 114 that latches the match results of a comparison operation. An individual match line will indicate a match only if all of the CAM cells 202 (actually compared) in that row match the comparand data. The latch results are then provided to priority encoder 116 which generates an address corresponding to at least one of the matched locations. For one embodiment, the matched address is the highest priority match address. The highest priority match address may be the lowest numbered address, the highest numbered address, or any other selected address. Alternatively, the match address may be the lowest priority match address, or any other predetermined priority.

[0036] Each column of CAM cells is coupled to a bit line (BL) 210, a complementary bit line (BLB) 212, a compare line (CL) 214, and a complementary compare line (CLB) 216. BL 210 and BLB 212 are coupled to sense amplifiers 122 that may enable data to be read from or written to CAM cells 202. CL 214 and CLB 216 are coupled to comparand register 120 and provide comparand data to CAM cells 202 for comparison purposes. For alternative embodiments, any other CAM array architecture may be used. For example, CAM array 102 may not include CL 214 and CLB 216; rather, BL 210 and BLB 212 may be coupled to comparand register 120 and may be used to perform a comparison with data stored in CAM cells 202 as generally known in the art. For example, in the first part of a compare cycle, compare data may be provided onto BL 210 and BLB 212 from comparand register 120. In the second part of the compare cycle, BL 210 and BLB 212 may be driven with data to be output from CAM array 104.

[0037] Figure 3 shows CAM cell 300 that is one embodiment of a CAM cell 202. CAM cell 300 includes comparator 302 and RAM cell 304. RAM cell 304 is coupled to BL 210, BLB 212, and word line 208, and may be any type of RAM cell. When RAM cell 304 is selected by word line 208, data may be read from or written to RAM cell 304 via bit lines BL 210 and BLB 212. Comparator 302 is coupled to RAM cell 304, CL 214, CLB 216, and match line 204. Comparator 302 may compare data from comparand register 120 (supplied on CL 214 and CLB 216) with data stored in RAM cell 304 (supplied on lines 218 and 220). Comparator 302 may output the comparison result to match line 204. Comparator 302 may be any type of comparison circuit including an exclusive OR (XOR) or exclusive NOR (XNOR) gate. The comparison and read

functions of CAM cell 300 may be performed simultaneously or sequentially. In alternative embodiments (e.g., CL 214 and CLB 216 omitted), the comparison and read functions may be performed sequentially. It will be appreciated that numerous other CAM cells may be used for CAM cells 202.

[0038] Figure 4 shows CAM cell 400 that is one embodiment of CAM cell 300 of Figure 3 and/or one embodiment of a CAM cell 202 of Figure 2. It will be appreciated that other CAM cells may be used to form CAM cell 300 and/or CAM cell 202. CAM cell 400 includes a RAM cell 404 and a comparator 402. RAM cell 404 includes cross-coupled inverters 406 and 408 coupled to nodes 428 and 430. Inverters 406 and 408 may be any type of inverters including NMOS, PMOS, or CMOS inverters with active or passive loads. RAM cell 404 also includes pass gates 410 and 412. Pass gate 410 may be an NMOS transistor having its source (drain) coupled to node 428, its drain (source) coupled to BL 210, and its gate coupled to word line 208. Pass gate 412 may be an NMOS transistor having its source (drain) coupled to node 430, its drain (source) coupled to BLB 212, and its gate coupled to word line 208. When word line 208 is selected (e.g., pulled to a high logic state), pass gate 410 is enabled to transfer data between node 428 and BL 210, and pass gate 412 is enabled to transfer data between node 430 and BLB 212.

[0039] Comparator 402 includes transistors 414, 416, 418, and 420. Transistor 414 has its gate coupled to node 428, its source coupled to line 426, and its drain coupled to the source of transistor 416. Transistor 416 has its drain coupled to match line 204, and its gate coupled to CLB 216. Transistor 420 has its gate coupled to node 430, its source

coupled to line 426, and its drain coupled to the source of transistor 418. Transistor 418 has its drain coupled to match line 204, and its gate coupled to CL 214. Line 426 may be driven to a voltage VREFL of approximately one threshold voltage above ground (e.g., from approximately 0.6 volts to approximately 0.8 volts) due to the diode configuration of NMOS transistor 422 having its source coupled to ground, and its drain and gate coupled to line 426. One transistor 422 may be used for more than one CAM cell in CAM array 104. Alternatively, VREFL may be driven to approximately zero volts to approximately one volt by a reference voltage generator or other circuitry. For an alternative embodiment, line 426 may be directly connected to ground.

[0040] CAM cell 400 may also include PMOS transistor 424 that may pull match line 204 towards VDD when there is no match between data stored in RAM cell 404 and comparand data provided on compare lines CL 214 and CLB 216. PMOS transistor 424 has its source coupled to VDD, its drain coupled to match line 204, and its gate coupled to a reference voltage VREFH. VREFH may be approximately one threshold drop below VDD (e.g., approximately 0.6 to 0.8 volts below VDD). Alternatively, VREFH may be other voltages. VDD may be any supply voltage. For example, VDD may be from approximately 2.7 volts to approximately 7.0 volts.

[0041] The operation of CAM cell 402 may be illustrated as follows. Assume that RAM cell 404 stores a high logic state (a one) at node 428 causing transistor 414 to be on. The cross-coupled nature of inverters 406 and 408 will cause a low logic state (a zero) to be stored at node 430 turning off transistor 420. During a compare operation, if the comparand data is also high (CL 214 high), then CLB 216 will be driven low causing



transistor 416 to turn off. Therefore, if there is a match between the comparand data and the data stored by RAM cell 404, match line 204 will remain in a high state. If, however, the comparand data is low (CL 214 low), then CLB 216 is driven high causing transistor 416 to be on and match line 204 to be pulled towards the potential of line 426 via transistors 414 and 416. Therefore, if there is no match between the comparand data and the data stored by RAM cell 404, match line 204 will be pulled to a low state.

[0042] Similarly, if RAM cell 404 stores a low logic state at node 428, transistor 414 will be off, and transistor 420 will be on. During a compare operation, if the comparand data is also low, then CL 214 will be driven low causing transistor 418 to turn off.

Therefore, if there is a match between the comparand data and the data stored by RAM cell 404, match line 204 will remain in a high state. If, however, the comparand data is high, then CL 214 is driven high causing transistor 418 to be on and match line 204 to be pulled towards the potential of line 426 via transistors 418 and 420. Therefore, if there is no match between the comparand data and the data stored by RAM cell 404, match line 204 will be pulled to a low state.

[0043] Because of the separate bit lines and compare lines, CAM cell 400 may perform a comparison operation at the same time that data is read from RAM cell 404. That is, data may be compared on lines CL 214 and CL 216 with nodes 428 and 430 at the same time that data is read from nodes 428, and 430 to BL 210 and BLB 212, respectively. This may be advantageous in performing compare operations in CAM devices such as CAM device 100 of Figure 1.

[0044] With respect to Figure 2, CAM array 104 may store associative memory data

or compare data (e.g., port address, pattern recognition keys, etc.) in any number of CAM cells 202 in a given row of CAM cells. The compare data may be compared with comparand data stored in comparand register 120. Similarly, CAM array 104 may store associated data or RAM data (e.g., bridge or switch port address, access information, or aging data) in any number of CAM cells 202 in a given row of CAM cells. The associated data may correspond to, or be associated with, other CAM cells in the same row that store compare data. The CAM cells 202 may be partitioned with a granularity of one into compare data and associated data segments. The compare data and associated data segments may be interleaved, or they may be non-interleaved segments.

**[0045]** For one embodiment, memory configuration register 106 may be used to program which bits in CAM array 104 are used to store compare data, and which bits are used to store associated data or other information. Memory configuration register 106 may thus act as a mask register indicating which CAM cells 202 (e.g., that may store compare data) will participate in a compare operation with comparand data, and which bits (e.g., that may store associated data or any other information) will not participate in a comparison operation with comparand data. For example, memory configuration register 106 may include one or more programmable bits corresponding to each column of CAM cells in CAM array 104. Each bit may be programmed via CBUS 138.

**[0046]** Data stored in CAM array 104 may be output (e.g., via sense amplifiers 122 and output multiplexer 134) in any order or in any format. For one embodiment, CAM array 104 may output its contents in multi-bit segments. Each segment may store compare data, associated data, and/or other information. For one example, CAM array

104 may be partitioned into four segments of 16 bits each. One or more of the segments may be sensed and output by output multiplexer 134 in any order (e.g., 24 or 16 possible ordered outputs). One or more bits in device configuration register 136 may be programmed via CBUS 138 to cause one or more multiplexers in output multiplexer 134 to output the desired order of the segments to ADS BUS 142. For example, programming a bit in device configuration register 136 may cause a first 16-bit segment to be output on the first sixteen lines of ADS BUS 142, and a second 16-bit segment to be output on the second sixteen lines of ADS BUS 142. For this example, the remaining signal lines of ADS BUS 142 may output a match address and status information as shown in Figure 10.

[0047] For another example, programming a bit in device configuration register 136 may cause a third 16-bit segment to be output on the first sixteen lines of ADS BUS 142, and a fourth 16-bit segment to be output on the second sixteen lines of ADS BUS 142.

[0048] Figure 5 describes one embodiment 500 of generally performing a write and compare instruction in CAM device 100 in one clock cycle of external clock signal CLK 178. At the start of the clock cycle, instruction decoder 128 decodes the write and compare instruction on IBUS 140 at step 502. In response to the instruction, comparand register 120 loads comparand data from CBUS 138 at step 504. Instruction decoder 128 may include a look-up table, a state machine, or any other control logic that can decode the write and compare or a compare instruction on IBUS 140.

[0049] At step 506, the comparand data in comparand register 120 is compared with data stored in CAM array 104 to determine if there is a match. The write and compare

instruction decoded by instruction decoder 128 may instruct CAM device 100 to compare the comparand data against all entries, only valid entries (e.g., skip and empty bits inactive), entries where the skip bits are active, or entries where the skip bit and empty bits are active. The write and compare instruction may also instruct CAM device 100 to use one or more mask registers (not shown) to mask certain columns of CAM cells from the comparison. Alternatively, the decoded write and compare instruction may instruct CAM device 100 to compare the comparand data against associated data stored in CAM array 104. The write and compare instruction may be binary encoded on IBUS 140, or may be encoded in any other format. The comparison results of step 506 will be reflected by the state of the match lines 204 in CAM array 104.

**[0050]** At step 508, the states of the match lines in CAM array 104 are latched by match latch 114 and provided to priority encoder 116. At step 510, priority encoder 116 determines the highest priority match address from the comparison. Match latch 114 may be incorporated into priority encoder 116. The match address is then output by priority encoder 116 to bus 146 and may be latched by address latch 118. Bus 146 couples the match address to output multiplexer 134 and status register 132 for output to ADS BUS 142. As generally known in the art, priority encoder 116 may also generate a match flag signal on line 148 if there is at least one match between the comparand data and the data stored in CAM array 104. Additionally, priority encoder 116 may also generate a multiple match flag signal on line 150 if there is at least two matches between the comparand data and the data stored in CAM array 104.

**[0051]** The match address on bus 162 is provided to address decoder 112 by address

selector 110. Address selector 110 couples either the address on address bus 164 or the match address on bus 162 to address decoder 112 in response to a signal on line 166. Address bus 164 may be an internal bus within CAM device 100 (e.g., coupled to an address counter) or a bus external to CAM device 100. Address selector 110 may be a multiplexer. For an alternative embodiment, address selector 110 may be omitted and bus 162 may be directly connected to address decoder 112.

**[0052]** At step 512, address decoder 112 decodes the match address and selects one row of CAM cells in CAM array 104 corresponding to the match address. Address decoder 112 may also select the corresponding validity bits 108 for the selected row in CAM array 104. At step 514, one or more CAM cells at the highest priority match address are sensed by sense amplifiers 122 and coupled to bus 152 for output to ADS BUS 142 via output multiplexer 134. Additionally, the validity bits may be accessed and/or sensed (e.g., by sense amplifiers 122) and output to bus 157. At step 516, and before the end of the first clock cycle of external clock CLK 178, output multiplexer 134 may simultaneously or concurrently output to ADS BUS 142: the data read from CAM array 104 onto bus 152; the match address on bus 146; and/or, the status information (e.g., match flag signal on line 148, multiple match flag signal on line 150, full flag signal on line 156, the validity bits on bus 157, and/or device identification information from device identification register 121 on bus 155). For example, output multiplexer 134 may output the signals as illustrated in Figures 10 and 11.

**[0053]** Timing generator 126 outputs timing signals on lines 160, 166, 168, 170, 172, 174, 176, and 182 in response to one or more clock signals from clock buffer 124 and an

indication of the write and compare instruction (or simply a compare instruction) from instruction decoder 128 on bus 159. The timing signals coordinate the operation of the various circuit elements to perform the write and compare instruction in one clock cycle.

**[0054]** Figure 6 is an illustrative timing diagram showing one embodiment of the sequence of signals generated by timing generator 126 for a write and compare instruction. At time  $t_0$ , the start of the clock cycle of CLK 178, instruction decoder 128 decodes the write and compare instruction on IBUS 140 and sends a signal on bus 159 to timing generator 126. In response, timing generator 126 sends a pulse signal on line 160 to cause the comparand data on CBUS 138 to be loaded into comparand register 120.

**[0055]** Between times  $t_0$  and  $t_1$ , the comparand data is provided to CAM array 104 and compared against one or more CAM cells of CAM array 104 for a match. The match results reflected on the match lines of CAM array 104 are then coupled to match latch 114 and latched by the pulse generated at time  $t_1$  on line 172. The latched match signals are coupled to priority encoder 116. At time  $t_2$ , timing generator 126 generates a pulse on line 174 that causes priority encoder 116 to generate a match address for the highest priority matched location. The highest priority match address is output to bus 146. At time  $t_3$ , timing generator 126 generates a pulse signal on line 176 that causes address latch 118 to latch the match address from priority encoder 116 and provide the latched match address to bus 162.

**[0056]** At time  $t_4$ , timing generator 126 generates a pulse signal on line 166 that causes optional address selector 110 to couple the match address on bus 162 to address decoder 112. Address decoder 112 may then decode the match address and select the row

of CAM cells in CAM array 104 and/ or the validity bits 108 that corresponds to the matched address. Address decoder 112 may alternatively be synchronous and start decoding the match address in response to the signal on line 166. Alternatively, address decoder 112 may start decoding the match address in response to a pulse signal on line 168. The pulse signal on line 168 may be generated coincident with the pulse signal on line 166, or after the pulse signal on line 166 but before the pulse signal on line 170.

**[0057]** Between times t4 and t5, sense amplifiers 122 sense the selected CAM cells of CAM array 104 (and/ or the corresponding validity bits). At time t5, timing generator 126 then generates a pulse signal on line 170 that causes sense amplifiers 122 to latch the sensed data and couple this data to bus 152. This data may then be output by output multiplexer 134 to ADS BUS 142 by time t6 along with the match address from bus 146 and the status information (e.g., the validity bits 108, the match flag signal from line 148, the multiple match flag signal from line 150, the full flag signal on line 156, and/or the device identification on bus 155). A subsequent instruction (e.g., another write and compare instruction) may then begin at time t6.

**[0058]** Timing generator 126 may additionally generate one or more pulse signals on bus 182 at time t6 that causes status register 132 to load the match address from bus 146, the match flag signal from bus 148, the multiple match flag signal from line 150, the device identification information from bus 155, the sensed data CAM array data on bus 152, the full flag signal from line 156, and/ or validity bits 108 from bus 157. The signals on bus 182 may alternatively be generated by instruction decoder 128.

**[0059]** The time differences between the pulses generated on lines 160, 172, 174,

176, 166, 170, and 182 are sufficient to allow the operations described above to function properly under anticipated, characterized, or specified process, temperature, and supply voltage ranges. For one embodiment, the clock cycle time (i.e.,  $t_6-t_0$ ) is approximately 25 ns,  $t_1-t_0$  is approximately 4 ns,  $t_2-t_1$  is approximately 2 ns,  $t_3-t_2$  is approximately 5 ns,  $t_4-t_3$  is approximately 2 ns,  $t_5-t_4$  is approximately 4 ns, and  $t_6-t_5$  is approximately 3 ns. For other embodiments, the time differences between the pulse signals may be different values.

**[0060]** The signals generated on lines 160, 172, 174, 176, 166, 170, and 182 are shown in Figure 6 as pulse signals. For alternative embodiments, a rising or falling edge of a signal may be generated instead of a pulse signal. The pulse signals may also be programmable or adjustable to have any pulse width.

**[0061]** Timing generator 126 may be any timing generator circuit that generates the pulses on lines 160, 172, 174, 176, 166, 170, and 182 at the appropriate times. Figure 7 shows timing generator 700 that is one embodiment of timing generator 126 of Figure 1. Timing generator 700 includes control logic 702 that receives one or more clock signals on bus 180 from clock buffer 124, and an indication of the write and compare instruction from instruction decoder 128 on bus 159. Control logic 702 may be a pulse generator circuit that generates a pulse signal on line 710 that may be coupled to line 160. The pulse signal on line 710 may be delayed by a series of delay elements 704, 705, 706, 707, 708, and 709 that may be used to generate the pulse signals on lines 172, 174, 176, 166, 170, and 182, respectively. Delay elements 704-709 may include, for example, resistors, capacitors, transistor pass gates, inverting buffers, and/ or non-inverting buffers. Each



delay element may also include a programmable element that enables a user or manufacturer to program the delay of each of the delay elements so that the pulses on lines 172, 174, 176, 166, 170, and 182 may be generated at the appropriate times. The programmable elements may also program the pulse width of one or more of the signals 172, 174, 176, 166, 170, and 182.

**[0062]** For an alternative embodiment, clock buffer 124 may generate a plurality of signals at different times, and timing generator 126 may include circuitry that selectively enables each of the clock signals to be coupled to lines 160, 172, 174, 176, 166, 170, and 182 at the appropriate time.

**[0063]** For still another embodiment, clock buffer 124 may generate a clock signal on bus 180 that has a frequency that is  $n$  times the frequency of CLK 178, where  $n$  is greater than one. Timing generator 126 may be a state machine or counter that consecutively generates the signals on lines 160, 172, 174, 176, 166, 170, and 182 once per clock cycle of the clock signal on bus 180.

**[0064]** Figure 8 illustrates timing generator 800 that is another embodiment of timing generator 126 of Figure 1. Timing generator 800 includes a clock generator 802 and latches 803-808. Clock buffer 802 generates one or more clock signals on bus 816 in response to one or more buffered clock signals received on bus 180 from clock buffer 124. Alternatively, clock generator 802 may receive CLK 178. The clock signals on bus 816 may be pulse signals or edge transitions, and may be of any pulse width. For one embodiment, the clock signals on bus 816 include pulse signals that are approximately 2 to 6 ns in duration. The clock signals from bus 816 are provided to the clock inputs of

latches 803-808. For example, one or more clock signals on lines 810, 811, 812, 813, 814, and 815 are coupled to the clock inputs of latches 803, 804, 805, 806, 807, and 808, respectively. The signals on lines 810-815 may be the same signal or different signals. For one embodiment, the signals on lines 810-815 may be pulse signals that start at the beginning of a clock cycle, the middle of clock cycle, at the end of a clock cycle, or at any other predetermined time within a clock cycle. Latches 803, 804, 805, 806, 807, and 808 may be self-timed latches that generate pulse signals on lines 172, 174, 176, 166, 170, and 182, respectively, in response to data input signals on lines 820-825 and the assertion of clock signals on lines 810-815. Lines 820-825 may be included in bus 159 output by instruction decoder 128 of Figure 1.

**[0065]** With respect to Figures 6 and 8, clock generator 802 may generate the signal on line 160 at time  $t_0$ . The signal on line 160 may alternatively be generated by a latch device, but it may be advantageous to generate this signal very close to the rising edge transition of CLK 178. For one embodiment, CLK 178 may be provided on line 160. After the write and compare instruction is decoded, instruction decoder 128 may assert signals on lines 820-825 that cause the pulses on lines 172, 174, 176, 166, 170, and 182 to be generated. Latch 803 may latch the signal on line 820 in response to a clock signal on line 810. The signal on line 810 may arrive at time  $t_1$ , or latch 803 may receive the signal on line 810 earlier and may delay the generation of the signal on line 172 until time  $t_1$ . Latch 804 may latch the signal on line 821 in response to a clock signal on line 811. The signal on line 811 may arrive at time  $t_2$ , or latch 804 may receive the signal on line 811 earlier and may delay the generation of the signal on line 174 until time  $t_2$ .

Latch 805 may latch the signal on line 822 in response to a clock signal on line 812. The signal on line 812 may arrive at time  $t_3$ , or latch 805 may receive the signal on line 812 earlier and may delay the generation of the signal on line 176 until time  $t_3$ . Latch 806 may latch the signal on line 823 in response to a clock signal on line 813. The signal on line 813 may arrive at time  $t_4$ , or latch 806 may receive the signal on line 813 earlier and may delay the generation of the signal on line 166 until time  $t_4$ . Latch 807 may latch the signal on line 824 in response to a clock signal on line 814. The signal on line 814 may arrive at time  $t_5$ , or latch 807 may receive the signal on line 814 earlier and may delay the generation of the signal on line 170 until time  $t_5$ . Latch 808 may latch the signal on line 825 in response to a clock signal on line 815. The signal on line 815 may arrive at time  $t_6$ , or latch 808 may receive the signal on line 815 earlier and may delay the generation of the signal on line 182 until time  $t_6$ . There may be one or more latches 808 to output one or more signals on one or more lines 182.

**[0066]** The previous embodiments have described a single clock cycle operation of a write and compare instruction. The CAM device 100 of Figure 1 may also perform a compare instruction in a single cycle. The compare instruction performs the same steps as the write and compare operation described above excluding the additional step of loading the comparand data into comparand register 120.

**[0067]** CAM device 100 may also function in a pipelined mode of operation to take advantage of the fact that CAM device 100 can perform a write and compare instruction (or simply a compare instruction) in a single clock cycle. An example of a pipelined mode of operation is illustrated in Figure 9. At time  $t_0$ , a first write and compare

instruction is provided to CAM device 100 on bus 140 at the same time that first comparand data is provide on CBUS 138 for comparand register 120. During clock cycle tcycle1, all of the steps of the process of Figure 5 and/or Figure 6 are performed by CAM device 100. By the end of tcycle1, status register 132 receives the match address on bus 146, the match flag signal on line 148, the multiple match flag signal on line 150, the full flag signal on line 156, the device identification information on bus 155, the data read from CAM array 104 on bus 152, and/ or validity bits 108 on bus 157. Status register 132 may be updated with this information by one or more signals on bus 182 from timing generator 126 or clock buffer 124.

**[0068]** The rising edge of CLK 178 at time t1 may clock the contents of status register 132 onto ADS BUS 142 via output multiplexer 134 at time t2. Output multiplexer 134 may be configured to output the data from status register 132 rather than bypass the status register 132 by programming one or more bits in configuration register 136. Figure 10 shows one embodiment of the data output on ADS BUS 142 from status register 132 via output multiplexer 134 when ADS BUS 142 is a 64-bit bus. Figure 11 shows another embodiment. For alternative embodiments, the data output from status register 132 may be in any other format or order. For yet other embodiments, there may be more than one status register 132 to further pipeline the data to be output to ADS BUS 142.

**[0069]** The time difference between times t1 and t2 may be very fast because it is the time to clock a register element. For one embodiment, the time difference between t1 and t2 may be approximately 2 -10 ns. Additionally, the data output to ADS BUS 142 from status register 132 may be valid for a large percentage of the clock cycle time (e.g.,

approximately 30 percent to approximately 90 percent) as the data will become invalid only at the end of the next clock cycle tcycle2.

**[0070]** At the end of clock cycle tcycle1 and the start of clock cycle tcycle2, a second write and compare instruction (or simply a compare instruction) may be provided on IBUS 140 along with second comparand data on CBUS 138. During clock cycle tcycle2, all of the steps of the process of Figure 5 and/or Figure 6 are performed by CAM device 100. By the end of tcycle2, status register 132 receives the match address on bus 146, the match flag signal on line 148, the multiple match flag signal on line 150, the full flag signal on line 156, the device identification information on bus 155, the data read from CAM array 104 via bus 152, and/ or validity bits 108 on bus 152. This information will then be available from status register 132 in clock cycle tcycle3.

**[0071]** It will be appreciated that the signals generated by timing generator 126 (e.g., the pulse signals on lines 160,172,174,176,166,170, and/or 182) may be generated in response to a transition of CLK 178 or another internal clock signal. It will also be appreciated that the signals generated by timing generator 126 may alternatively occur over more than one clock cycle of CLK 178. Preferably, the signals span less than three clock cycles of CLK 178. For example, in the first clock cycle of CLK 178 the pulses on lines 160, 172, 174 and/or 176 may be generated; and, in the second clock cycle of CLK 178 the pulses on lines 166, 170, and/or 182 may be generated. For yet another embodiment, CLK 178 may run at a higher or lower frequency than an internal clock signal that may be used to start the sequence of pulse signals output by timing generator 126.

[0072] In the foregoing specification the invention has been described with reference to specific exemplary embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. The specification and drawings are, accordingly, to be regarded in an illustrative rather than restrictive sense.